

Delicious Pizza for Everyone!

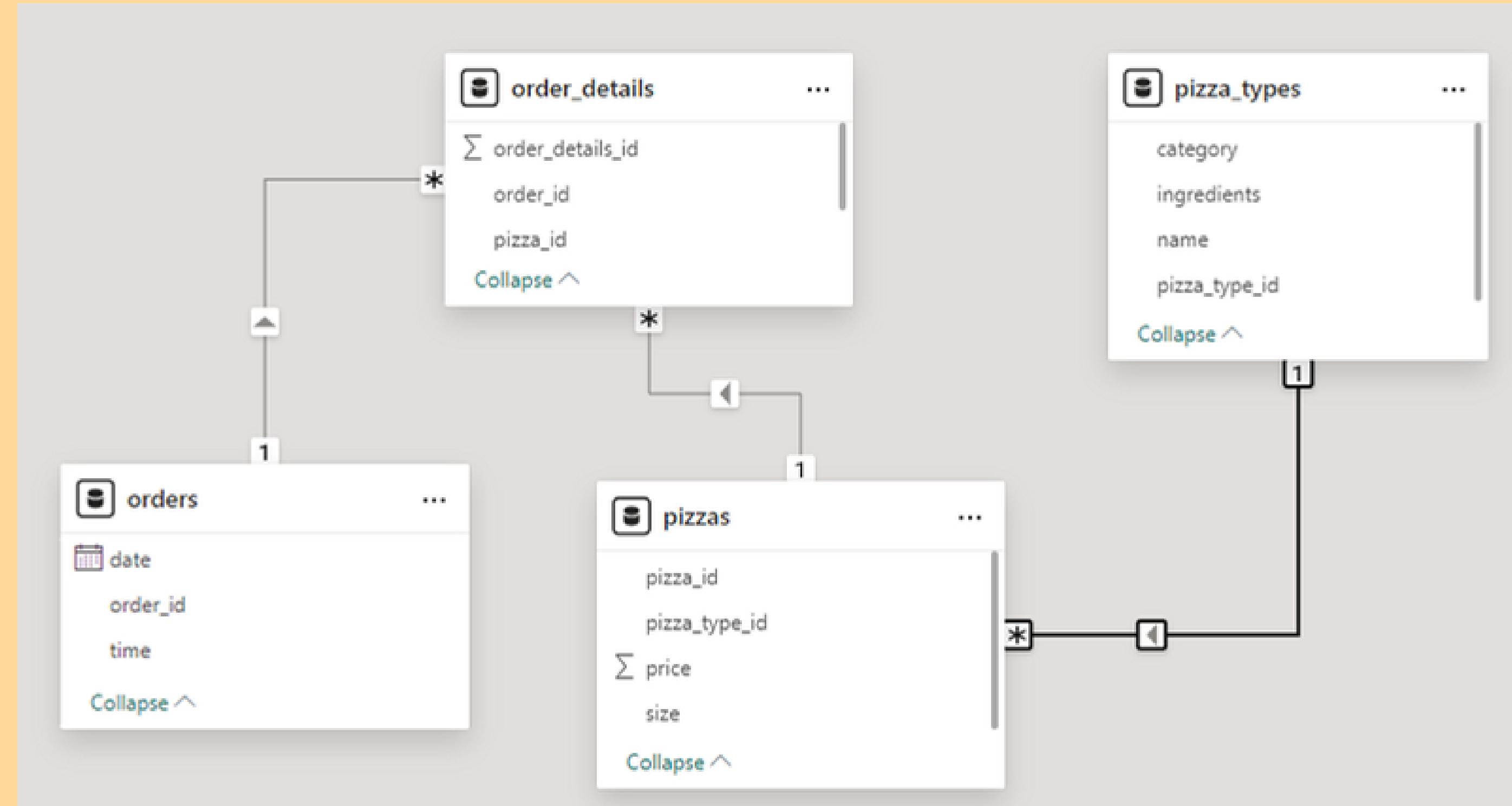
# SQL PROJECT ON PIZZA SALES

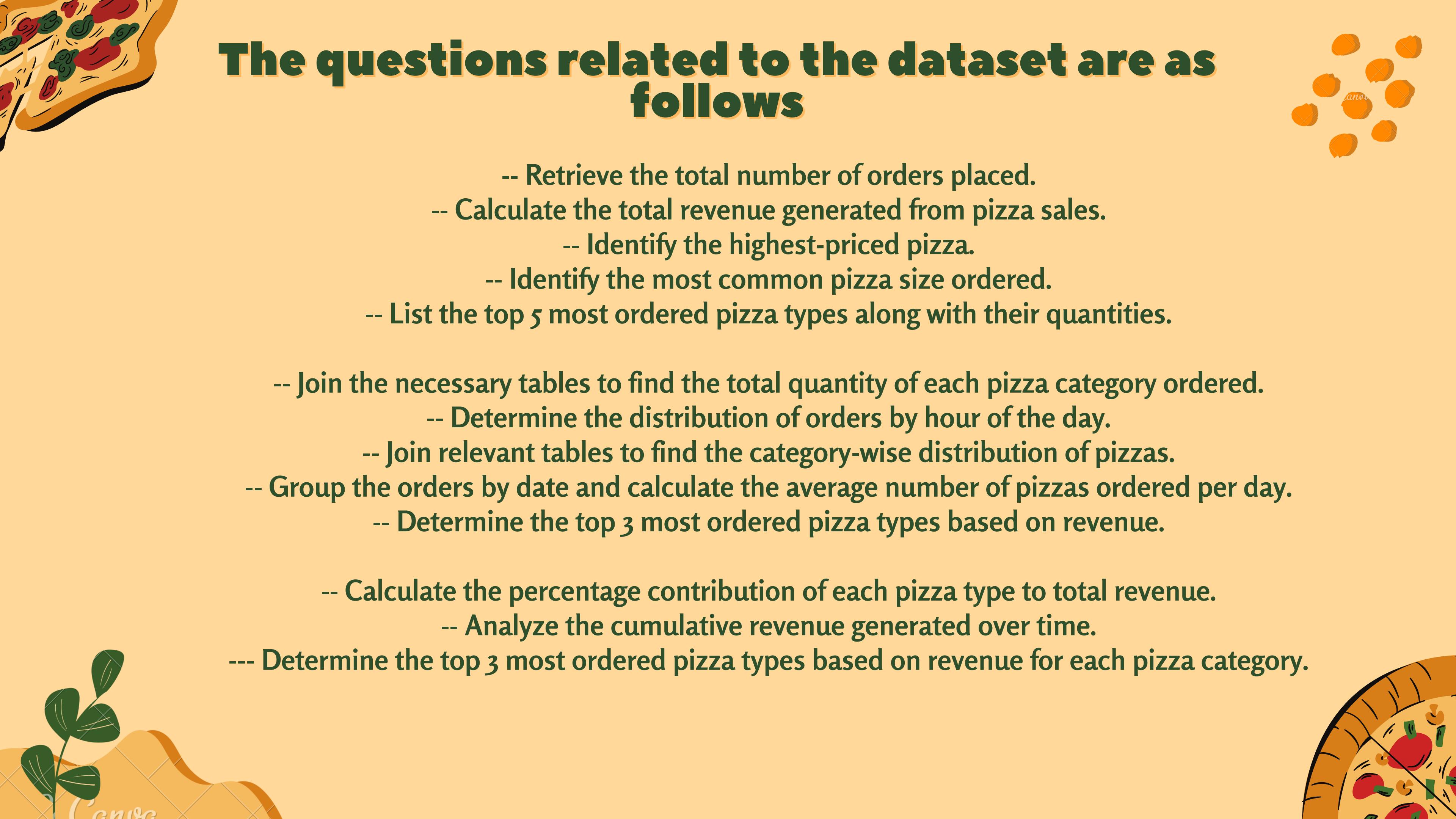


# Hello!

My name is Nikhil Kalia and in this project I have utilized  
SQL Queries to solve questions that were related to  
Pizza Sales.

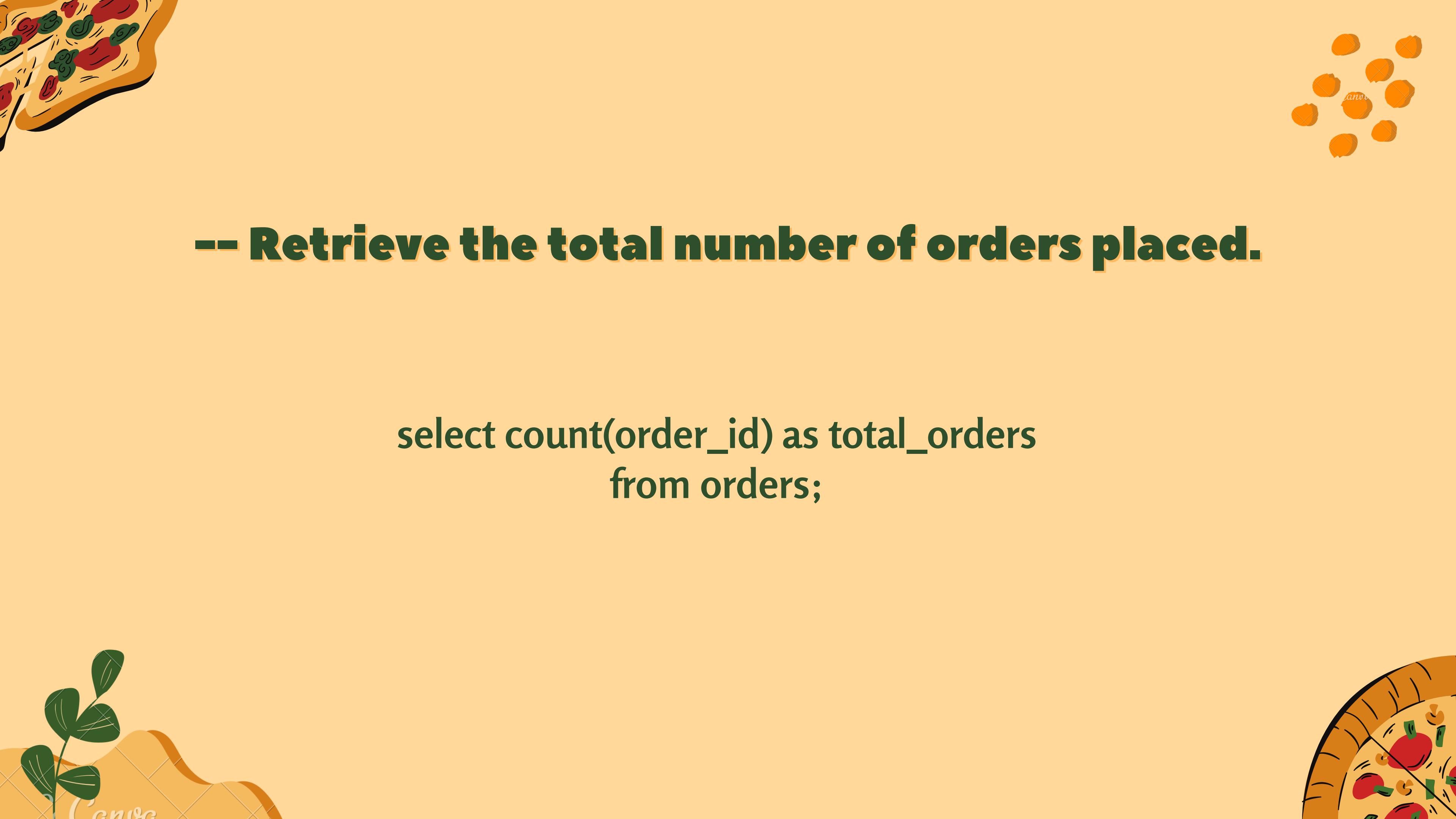
# Here is the schema of the dataset





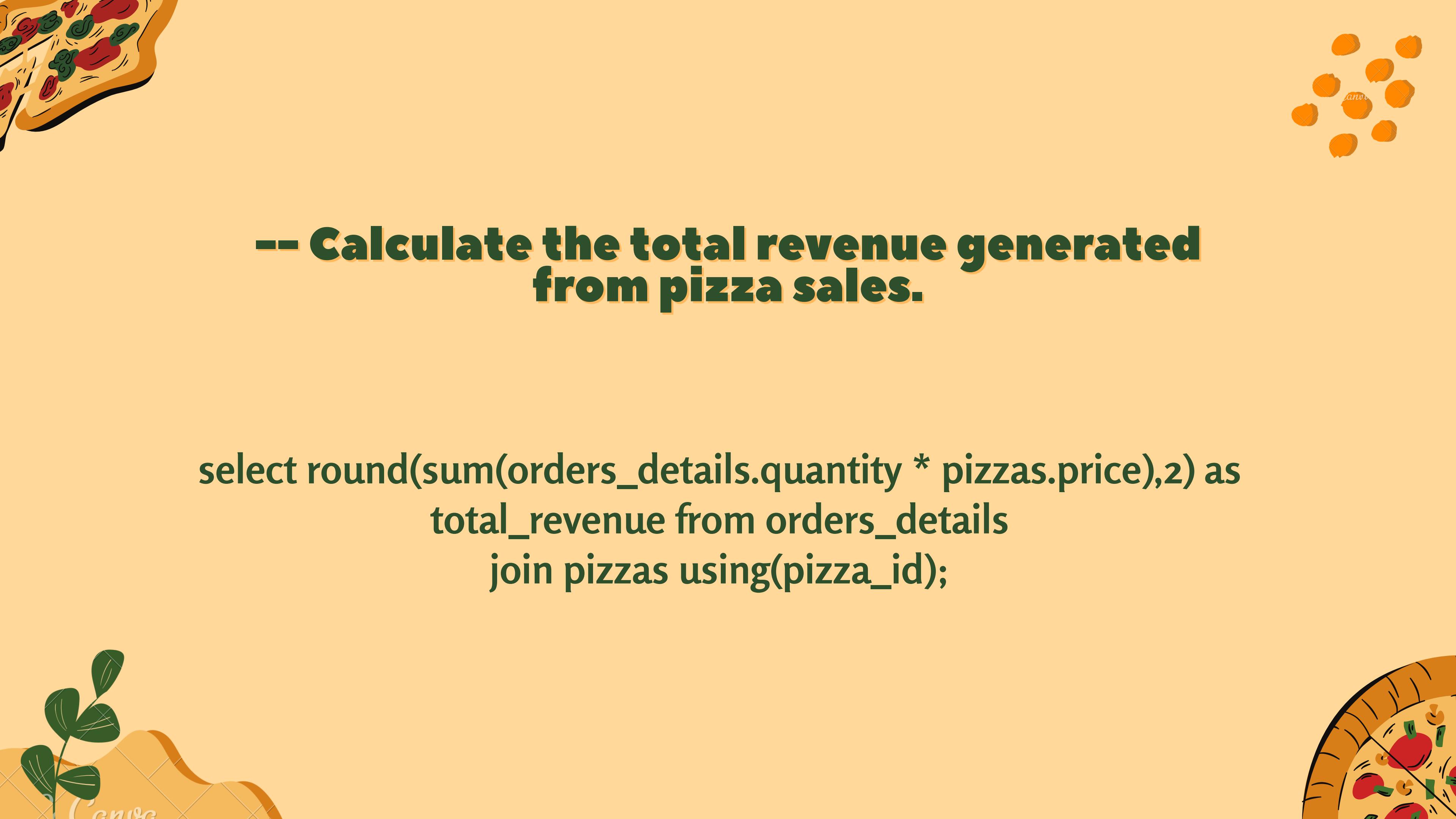
# The questions related to the dataset are as follows

- Retrieve the total number of orders placed.
- Calculate the total revenue generated from pizza sales.
  - Identify the highest-priced pizza.
  - Identify the most common pizza size ordered.
- List the top 5 most ordered pizza types along with their quantities.
  
- Join the necessary tables to find the total quantity of each pizza category ordered.
  - Determine the distribution of orders by hour of the day.
  - Join relevant tables to find the category-wise distribution of pizzas.
- Group the orders by date and calculate the average number of pizzas ordered per day.
  - Determine the top 3 most ordered pizza types based on revenue.
  
- Calculate the percentage contribution of each pizza type to total revenue.
  - Analyze the cumulative revenue generated over time.
- Determine the top 3 most ordered pizza types based on revenue for each pizza category.



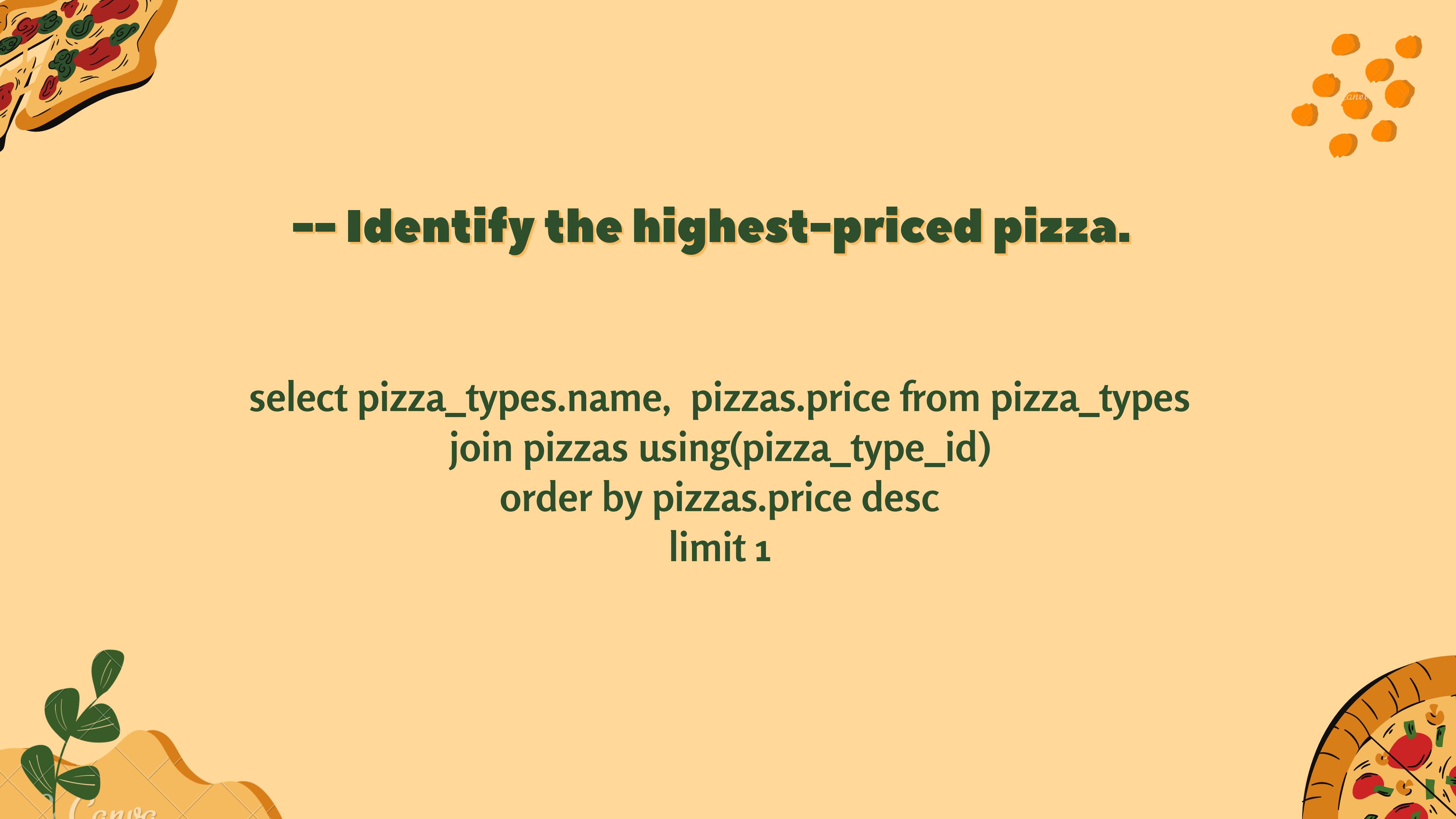
-- Retrieve the total number of orders placed.

```
select count(order_id) as total_orders  
from orders;
```



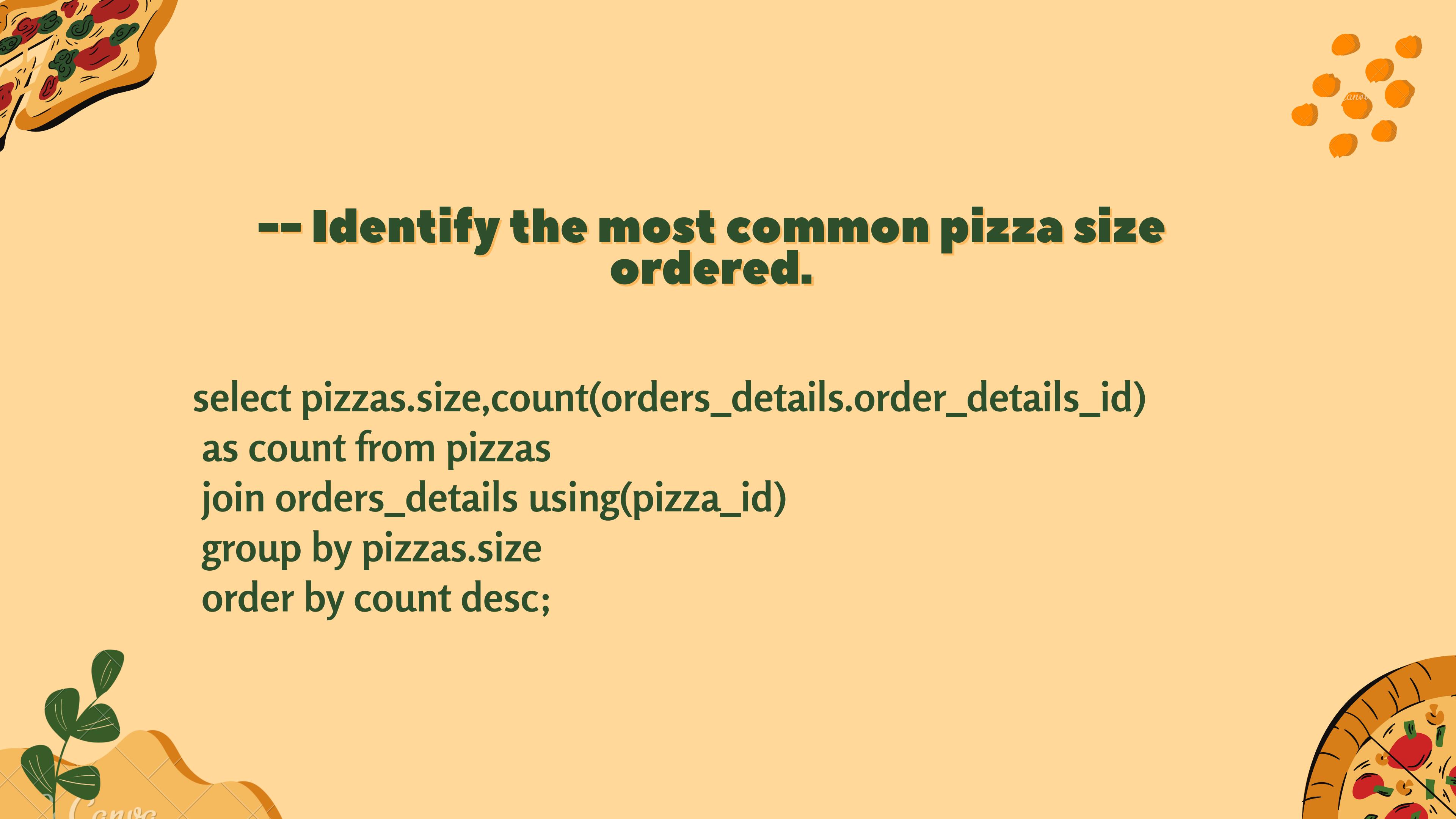
**-- Calculate the total revenue generated from pizza sales.**

```
select round(sum(orders_details.quantity * pizzas.price),2) as  
total_revenue from orders_details  
join pizzas using(pizza_id);
```



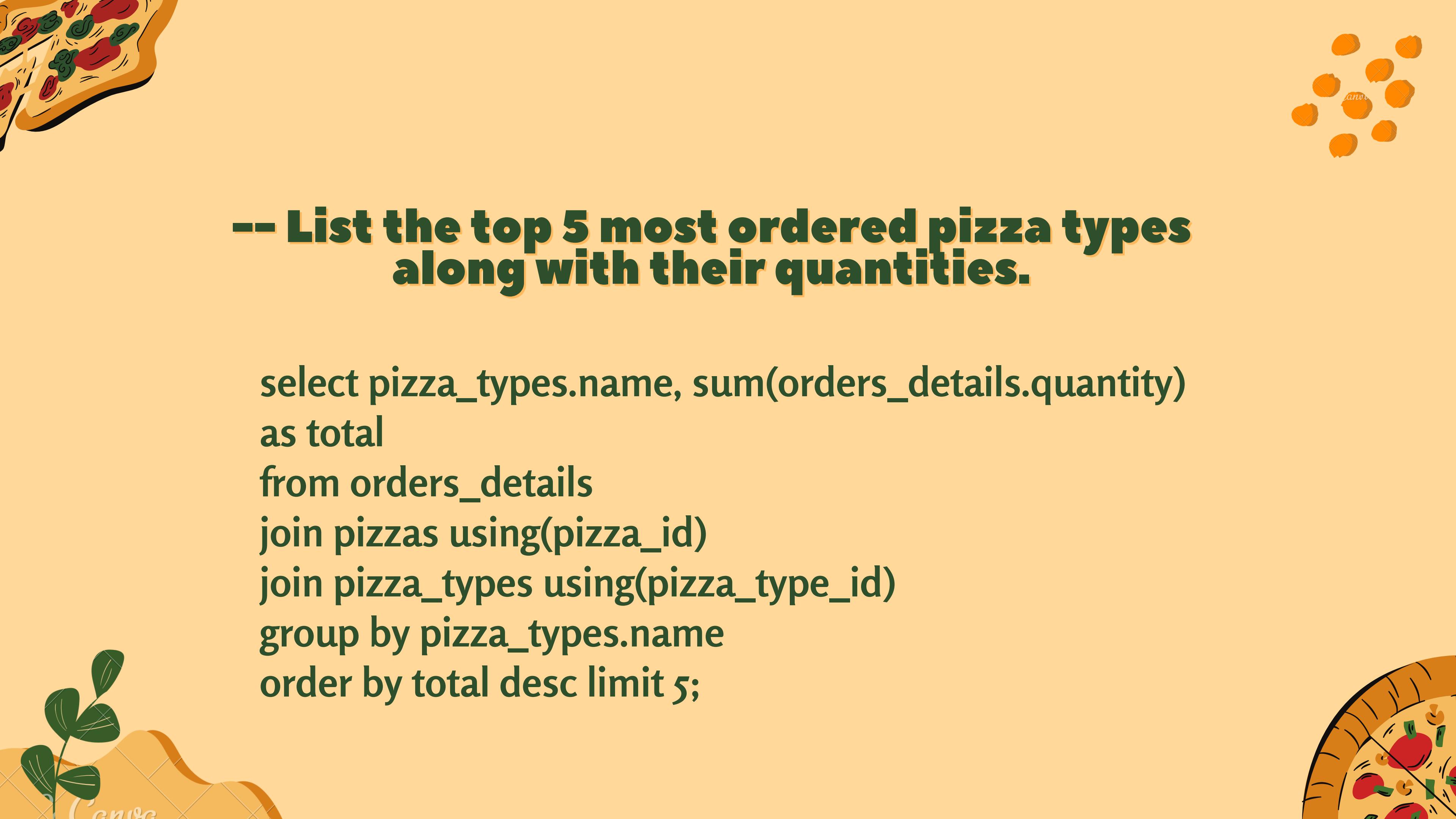
## -- Identify the highest-priced pizza.

```
select pizza_types.name, pizzas.price from pizza_types  
join pizzas using(pizza_type_id)  
order by pizzas.price desc  
limit 1
```



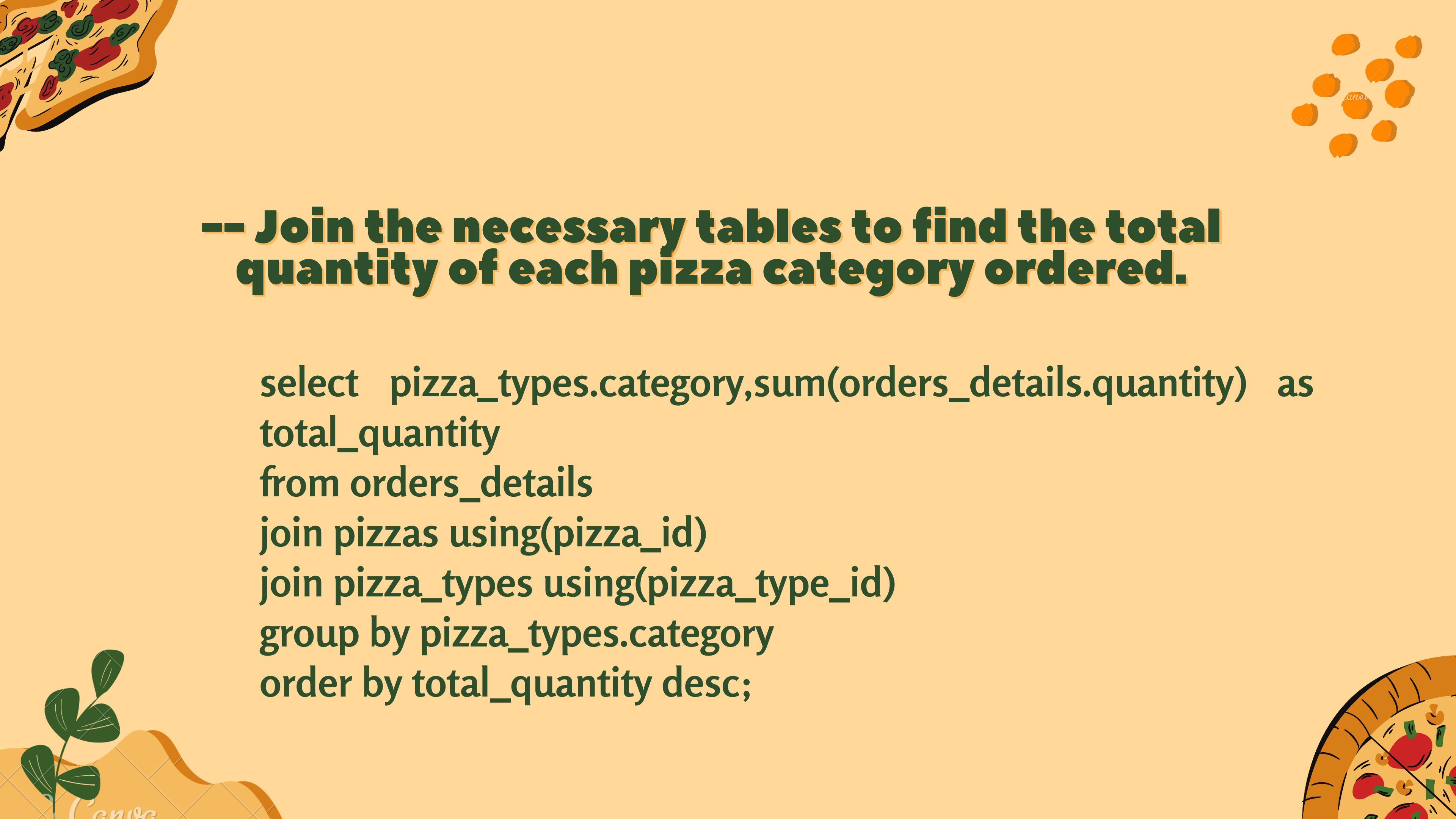
-- Identify the most common pizza size ordered.

```
select pizzas.size, count(orders_details.order_details_id)  
as count from pizzas  
join orders_details using(pizza_id)  
group by pizzas.size  
order by count desc;
```



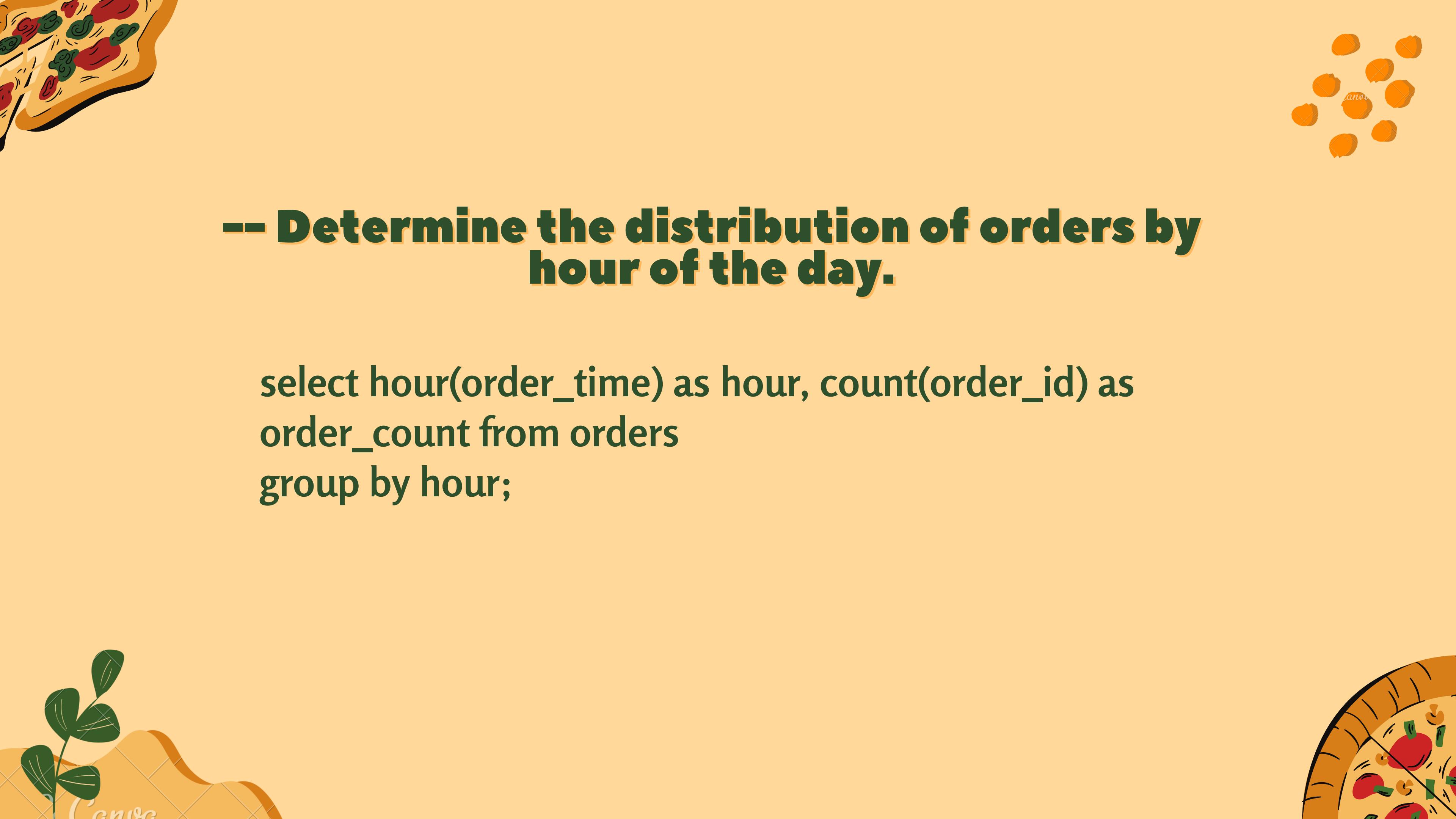
**-- List the top 5 most ordered pizza types along with their quantities.**

```
select pizza_types.name, sum(orders_details.quantity)
as total
from orders_details
join pizzas using(pizza_id)
join pizza_types using(pizza_type_id)
group by pizza_types.name
order by total desc limit 5;
```



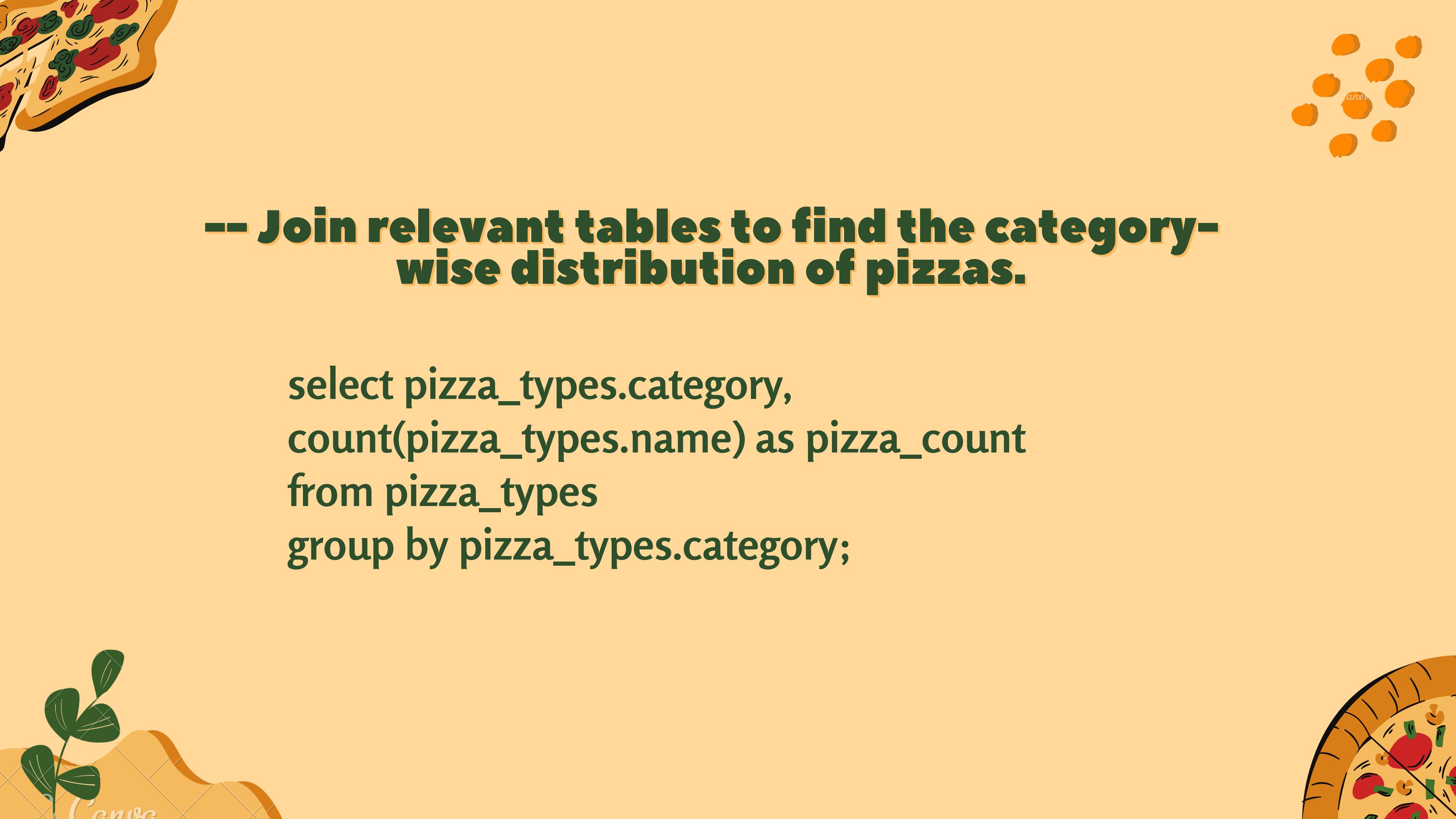
-- Join the necessary tables to find the total quantity of each pizza category ordered.

```
select pizza_types.category,sum(orders_details.quantity) as  
total_quantity  
from orders_details  
join pizzas using(pizza_id)  
join pizza_types using(pizza_type_id)  
group by pizza_types.category  
order by total_quantity desc;
```



**-- Determine the distribution of orders by hour of the day.**

```
select hour(order_time) as hour, count(order_id) as  
order_count from orders  
group by hour;
```

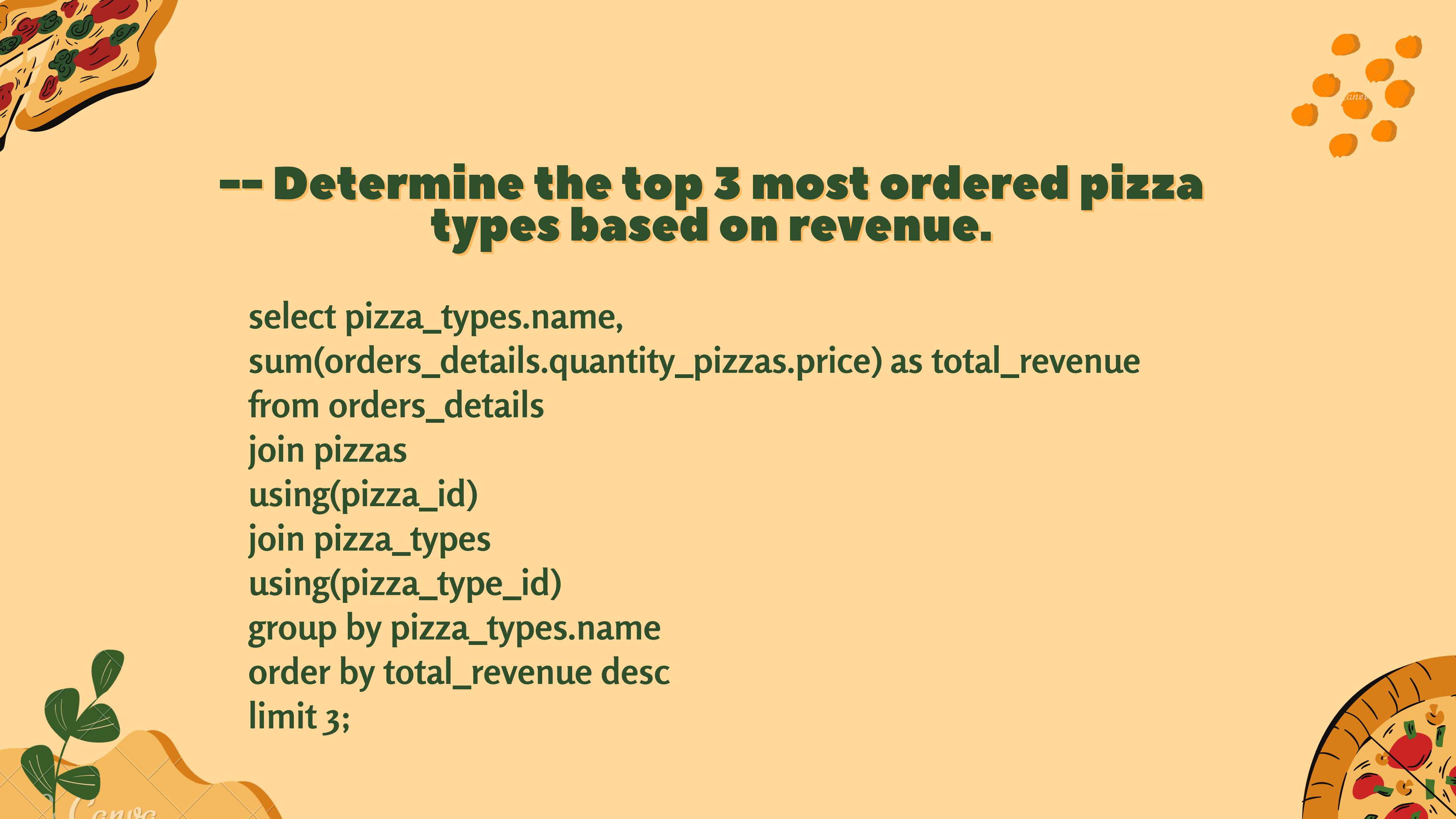


-- Join relevant tables to find the category-wise distribution of pizzas.

```
select pizza_types.category,  
       count(pizza_types.name) as pizza_count  
  from pizza_types  
 group by pizza_types.category;
```

**-- Group the orders by date and calculate the average number of pizzas ordered per day.**

```
select round(avg(total_quantity),0) as avg_ordered_per_day  
from(  
    select orders.order_date,  
        sum(orders_details.quantity)as total_quantity  
    from orders  
    join orders_details  
    using(order_id)  
    group by order_date) as a;
```

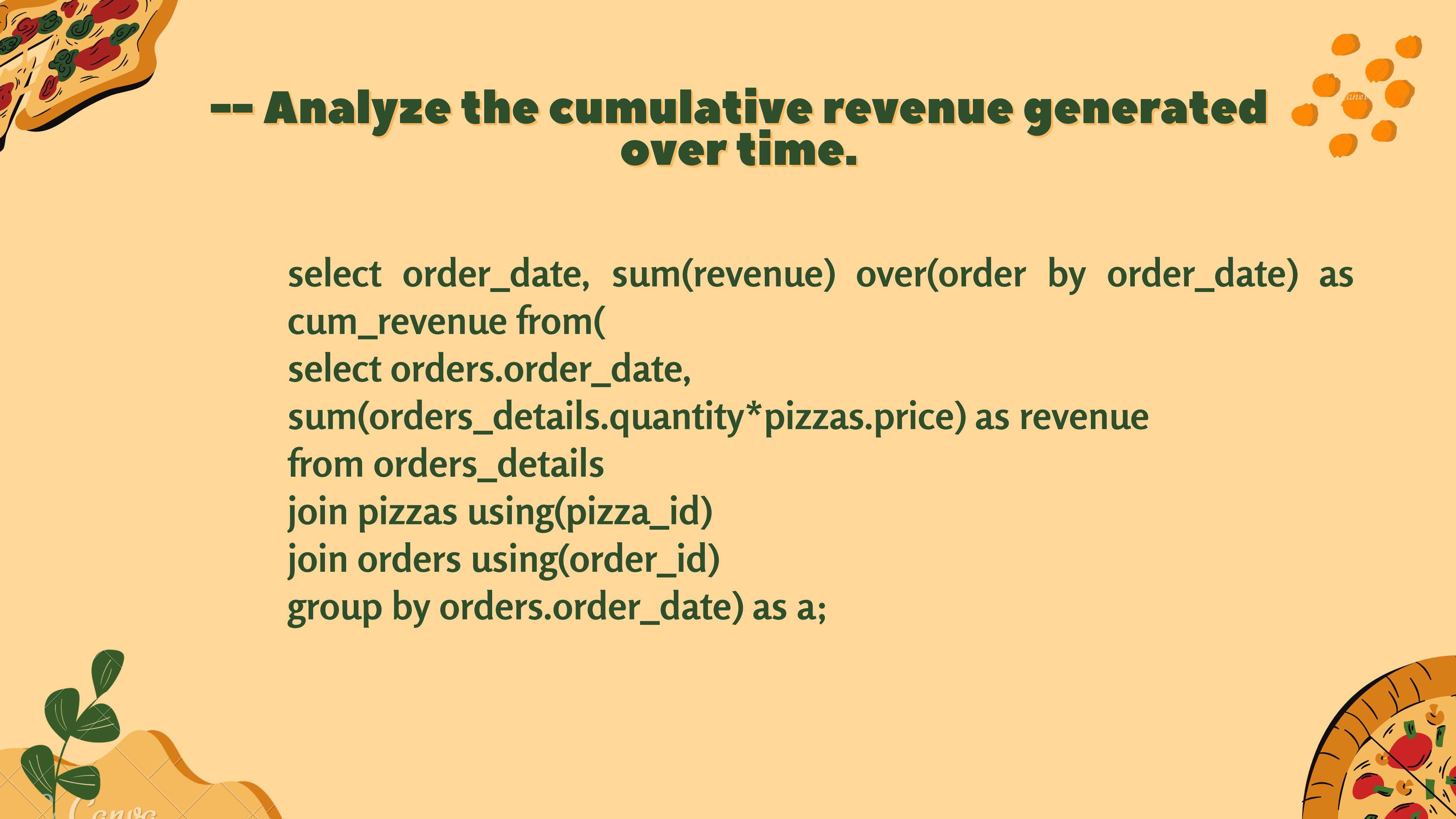


**-- Determine the top 3 most ordered pizza types based on revenue.**

```
select pizza_types.name,  
sum(orders_details.quantity_pizzas.price) as total_revenue  
from orders_details  
join pizzas  
using(pizza_id)  
join pizza_types  
using(pizza_type_id)  
group by pizza_types.name  
order by total_revenue desc  
limit 3;
```

**-- Calculate the percentage contribution of each pizza category to total revenue.**

```
select pizza_types.category,  
concat(round(sum(orders_details.quantity*pizzas.price)/(select  
round(sum(orders_details.quantity*pizzas.price),2)  
from orders_details join pizzas  
using(pizza_id))*100,2),"%) as revenue  
from pizza_types join pizzas  
using(pizza_type_id)  
join orders_details using(pizza_id)  
group by pizza_types.category  
order by revenue desc;
```



## -- Analyze the cumulative revenue generated over time.

```
select order_date, sum(revenue) over(order by order_date) as  
cum_revenue from(  
select orders.order_date,  
sum(orders_details.quantity*pizzas.price) as revenue  
from orders_details  
join pizzas using(pizza_id)  
join orders using(order_id)  
group by orders.order_date) as a;
```

**-- Determine the top 3 most ordered pizza types based on revenue for each pizza category.**

```
select category, name, revenue, rn
from(
select category,name,revenue, rank() over(partition by category
order by revenue desc) as rn
from(select
pizza_types.category,pizza_types.name,
sum(orders_details.quantity * pizzas.price) as revenue
from pizza_types
join pizzas using(pizza_type_id)
join orders_details using(pizza_id)
group by pizza_types.category,pizza_types.name) as a)b
where rn <= 3;
```

# THANK YOU



**nkalia87@gmail.com**



**[linkedin.com/in/nikhil-kalia-740b60242](https://www.linkedin.com/in/nikhil-kalia-740b60242)**